



Japan Patent Office

This is to certify that the annexed is a true copy of the following application as filed with this Office.

Date of Application: August 12, 2003

Application Number: Japanese Patent Application
No.2003-292492

[ST.10/C]: [JP2003-292492]

Applicant(s): RICOH COMPANY, LTD.

September 2, 2003

Commissioner,
Japan Patent Office

Yasuo Imai (Seal)

Certificate No.2003-3071728

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2003年 8月12日
Date of Application:

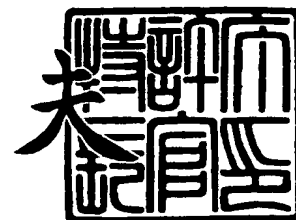
出願番号 特願2003-292492
Application Number:
[ST. 10/C]: [JP2003-292492]

出願人 株式会社リコー
Applicant(s):

2003年 9月 2日

特許庁長官
Commissioner,
Japan Patent Office

今井 康



出証番号 出証特2003-3071728

【書類名】 特許願
【整理番号】 0305955
【提出日】 平成15年 8月12日
【あて先】 特許庁長官 今井 康夫 殿
【国際特許分類】 G03G 21/00
【発明者】
 【住所又は居所】 東京都大田区中馬込 1 丁目 3 番 6 号 株式会社リコー内
 【氏名】 小林 寛樹
【特許出願人】
 【識別番号】 000006747
 【氏名又は名称】 株式会社リコー
【代理人】
 【識別番号】 100070150
 【弁理士】
 【氏名又は名称】 伊東 忠彦
【先の出願に基づく優先権主張】
 【出願番号】 特願2002-258190
 【出願日】 平成14年 9月 3日
【手数料の表示】
 【予納台帳番号】 002989
 【納付金額】 21,000円
【提出物件の目録】
 【物件名】 特許請求の範囲 1
 【物件名】 明細書 1
 【物件名】 図面 1
 【物件名】 要約書 1
 【包括委任状番号】 9911477

【書類名】 特許請求の範囲**【請求項 1】**

ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表す W e b ページを前記端末に送信する送受信制御手段と、

圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、

該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、

前記書式情報データを格納する第二の記憶手段と、

前記書式情報データを使用して前記 W e b ページを生成する W e b ページ生成手段とを有し、

該 W e b ページ生成手段により生成された前記 W e b ページを前記送受信制御手段により前記端末に送信することを特徴とする画像処理装置。

【請求項 2】

前記複数の書式情報ファイルは X S L ファイルであることを特徴とする請求項 1 記載の画像処理装置。

【請求項 3】

前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記 W e b ページの送信要求を受信するまでに、前記書式情報ファイルを解凍することを特徴とする請求項 1 又は 2 記載の画像処理装置。

【請求項 4】

前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記 W e b ページの送信要求を受信した際に、前記書式情報ファイルを解凍することを特徴とする請求項 1 又は 2 記載の画像処理装置。

【請求項 5】

前記解凍手段は、前記第一の記憶手段が格納する前記書式情報ファイルのうち、前記圧縮されている複数の書式情報ファイルの全てを解凍することを特徴とする請求項 3 又は 4 記載の画像処理装置。

【請求項 6】

前記解凍手段は、前記端末から W e b ページの送信要求を受信する際に、要求された前記 W e b ページを生成するために使用される前記書式情報ファイルを解凍することを特徴とする請求項 1 乃至 4 いずれか一項記載の画像処理装置。

【請求項 7】

前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記 W e b ページの送信要求を受信するより前又は受信した際に、前記第一の記憶手段が格納する前記書式情報ファイルのうち、使用された時期に近いものから順に所定の数の前記書式情報ファイルを解凍することを特徴とする請求項 6 記載の画像処理装置。

【請求項 8】

前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記 W e b ページの送信要求を受信するより前又は受信した際に、前記第一の記憶手段が格納する前記書式情報ファイルのうち、使用頻度が高いものから順に所定の数の前記書式情報ファイルを解凍することを特徴とする請求項 6 記載の画像処理装置。

【請求項 9】

前記 W e b ページ生成手段が前記 W e b ページを生成した後、前記 W e b ページの生成に使用された前記書式情報データは、前記第二の記憶手段から削除されることを特徴とする請求項 6 記載の画像処理装置。

【請求項 1 0】

前記 W e b ページ生成手段は、要求された前記 W e b ページの生成に使用される前記書式情報データが既に前記第二の記憶手段に格納されているかを判断する解凍済み判断手段を有し、

前記 W e b ページ生成手段は、前記解凍済み判断手段の判断結果に基づいて前記書式情

報データを使用して前記Webページを生成することを特徴とする請求項6乃至8いずれか一項記載の画像処理装置。

【請求項11】

前記Webページ生成手段は、前記書式情報データの数が所定の値を超えた場合は、いずれかの前記書式情報データを前記第二の記憶手段から削除することを特徴とする請求項10記載の画像処理装置。

【請求項12】

前記Webページ生成手段は、使用されていない期間が最も長い前記書式情報データを削除することを特徴とする請求項11記載の画像処理装置。

【請求項13】

前記Webページ生成手段は、解凍された時期が最も早い前記書式情報データを削除することを特徴とする請求項11記載の画像処理装置。

【請求項14】

ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する画像処理装置によるWebページ生成方法であって、

圧縮されている書式情報ファイルを書式情報データに解凍する書式情報解凍手順と、
前記書式情報データを使用してWebページを生成するWebページ生成手順とを有することを特徴とするWebページ生成方法。

【請求項15】

前記解凍手順は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するまでに実行されることを特徴とする請求項14記載のWebページ生成方法。

【請求項16】

前記解凍手順は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信した際に、実行されることを特徴とする請求項14記載のWebページ生成方法。

【請求項17】

前記解凍手順は、圧縮されている全ての前記書式情報ファイルを解凍することを特徴とする請求項15又は16記載のWebページ生成方法。

【請求項18】

前記解凍手順は、前記端末からWebページの送信要求を受信する際に、要求された前記Webページの生成に必要な前記書式情報ファイルを解凍することを特徴とする請求項14乃至16いずれか一項記載のWebページ生成方法。

【請求項19】

前記Webページ生成手順において、前記Webページを生成した後、

前記Webページの生成に使用された前記書式情報データは削除することを特徴とする請求項18記載の画像処理装置。

【請求項20】

要求された前記Webページの生成に使用される前記書式情報ファイルが既に前記書式情報データに解凍されているかを判断する解凍済み判断手順を有し、

前記Webページ生成手順は、前記解凍済み判断手順の判断結果に基づいて前記書式情報データを使用して前記Webページを生成することを特徴とする請求項18記載のWebページ生成方法。

【請求項21】

画像処理装置と端末とがネットワークを介して接続された画像処理システムであって、

前記画像処理装置は、前記端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、

圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、

該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報デ

ータに解凍する解凍手段と、
前記書式情報データを格納する第二の記憶手段と、
前記書式情報データを使用して前記W e b ページを生成するW e b ページ生成手段とを有し、
該W e b ページ生成手段により生成された前記W e b ページを前記送受信制御手段により前記端末に送信し、
前記端末は、前記W e b ページを表示する表示手段を有することを特徴とする画像処理システム。

【請求項 2 2】

前記複数の書式情報ファイルはX S L ファイルであることを特徴とする請求項 2 1 記載の画像処理装置システム。

【請求項 2 3】

ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表すW e b ページを前記端末に送信する送受信制御手段と、
圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、
該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、
前記書式情報データを格納する第二の記憶手段と、
前記書式情報データを使用して前記W e b ページを生成するW e b ページ生成手段とを有し、
該W e b ページ生成手段により生成された前記W e b ページを前記送受信制御手段により前記端末に送信することを特徴とする情報処理装置。

【請求項 2 4】

前記複数の書式情報ファイルはX S L ファイルであることを特徴とする請求項 2 3 記載の情報処理装置。

【請求項 2 5】

情報処理装置と端末とがネットワークを介して接続された情報処理システムであって、
前記情報処理装置は、前記端末からの要求を受信し、前記端末に表示される情報を表すW e b ページを前記端末に送信する送受信制御手段と、
圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、
該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、
前記書式情報データを格納する第二の記憶手段と、
前記書式情報データを使用して前記W e b ページを生成するW e b ページ生成手段とを有し、
該W e b ページ生成手段により生成された前記W e b ページを前記送受信制御手段により前記端末に送信し、
前記端末は、前記W e b ページを表示する表示手段を有することを特徴とする情報処理システム。

【請求項 2 6】

前記複数の書式情報ファイルはX S L ファイルであることを特徴とする請求項 2 5 記載の情報処理システム。

【書類名】明細書

【発明の名称】画像処理装置

【技術分野】

【0001】

本発明は、画像処理装置に関し、特にWebサーバの機能を有する画像処理装置に関する。

【背景技術】

【0002】

近年では、インターネット等のネットワークを介して交換する文書データは、その文書構造と書式情報とを分離させて扱うことが一般的になりつつある。文書データの文書構造と書式情報とを分離することで、コンピュータによる文書データの情報処理が容易となり、また一つの文書に様々な書式情報を与えてレイアウトし、再利用することができる。

【0003】

文書データを文書構造と書式情報とに分離した例としてXML (eXtensible Markup Language) データとXSL (eXtensible Stylesheet Language) データとによる構成が挙げられる。一般的なWebサーバにおいては、インターネット上に公開するWebページに掲載する情報としてXMLデータを生成し、あるいは保持する。Webサーバは、Webクライアントからの要求を受けて、当該XMLデータに対してXSLデータを適用してXSLT (XSL Transformations) 変換を行うことでHTML (HyperText Markup Language) データを生成し、当該HTMLデータをWebクライアントに送信する。

【0004】

一方、最近では画像処理装置においてもWebサーバとしての機能を有するものが存在し、例えば画像処理装置のステータスや、保管された文書情報、スプールされているジョブ、FAXやe-mail等に使用されるアドレス帳を汎用的なWebブラウザから見る事ができる。

【発明の開示】

【発明が解決しようとする課題】

【0005】

しかしながら、XSLデータを格納するXSLファイルはファイルサイズが大きくなりがちであり、画像処理装置のようにメモリ資源 (ROM) が限られている機器において種々のXSLファイルを保持しておくのは非常に困難であるという問題がある。

【0006】

即ち、一般的にXSLファイルのようなテキストファイルには、改行、タブ及びインデント等のように可読性を向上させるための制御コードが入力されることが多い。従って、XSLファイルは、必然的にファイルサイズが大きくなってしまふのである。

【0007】

XSLファイルによるメモリ資源の圧迫は、大容量のハードディスクを備えることにより解決することも可能であるが、その場合には、画像処理装置のコストアップにつながるという問題がある。

【0008】

本発明は、上記の点に鑑みてなされたものであって、書式情報を使用したWebページの生成を少ない記憶容量で行うことができる画像処理装置の提供を目的とする。

【課題を解決するための手段】

【0009】

そこで上記課題を解決するため、本発明は、請求項1に記載されるように、ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、前記書式情報データを格納する第二の記憶手段と、前記書式情報データを使用して前記Webページを生成するWebペ

ージ生成手段とを有し、該Webページ生成手段により生成された前記Webページを前記送受信制御手段により前記端末に送信することを特徴とする。

【0010】

このような画像処理装置では、Webページを生成するための書式情報を圧縮して記憶手段に格納しておくため、書式情報を使用したWebページの生成を少ない記憶容量で行うことができる。

【0011】

また、本発明は、請求項2に記載されるように、前記複数の書式情報ファイルはXSLファイルであることを特徴とする。

【0012】

このような画像処理装置では、書式情報の定義として広く利用されているXSL (eXtensible Stylesheet Language) を利用するため、Webページの生成処理の開発を効率的に行うことができる。

【0013】

また、本発明は、請求項3に記載されるように、前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するまでに、前記書式情報ファイルを解凍することを特徴とする。

【0014】

このような画像処理装置では、端末から最初のWebページの送信要求を受信するより前に予め書式情報ファイルを解凍しておくため、端末の要求に対するレスポンスの悪化を防止することができる。

【0015】

また、本発明は、請求項4に記載されるように、前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信した際に、前記書式情報ファイルを解凍することを特徴とする。

【0016】

このような画像処理装置では、端末から最初のWebページの送信要求を受信した際に書式情報ファイルを解凍するため、それまで解凍先の記憶領域の容量を節約できると共に、以降の端末の要求に対するレスポンスの悪化を防止することができる。

【0017】

また、本発明は、請求項5に記載されるように、前記解凍手段は、前記第一の記憶手段が格納する前記書式情報ファイルのうち、前記圧縮されている複数の書式情報ファイルの全てを解凍することを特徴とする。

【0018】

このような画像処理装置では、予め全ての書式情報ファイルを解凍しておく。従って、以降の端末からのWebページの送信要求に対応したWebページの生成に必要な書式情報ファイルは必ず解凍されており、端末の要求に対するレスポンスの悪化を防止することができる。

【0019】

また、本発明は、請求項6に記載されるように、前記解凍手段は、前記端末からWebページの送信要求を受信する際に、要求された前記Webページを生成するために使用される前記書式情報ファイルを解凍することを特徴とする。

【0020】

このような画像処理装置では、端末からWebページの送信要求を受信する際に、要求されたWebページの生成に必要な書式情報ファイルを解凍するため、解凍先の記憶領域の容量を節約することができる。

【0021】

また、本発明は、請求項7に記載されるように、前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するより前又は受信した際に、前記第一の記憶手段が格納する前記書式情報ファイルのうち、使用された時

期が近いものから順に所定の数の前記書式情報ファイルを解凍することを特徴とする。

【0022】

このような画像処理装置では、使用された時期が近いものから順に所定の数の書式情報ファイルを解凍しておくため、解凍先の記憶領域の容量を節約することができると共に、最近要求されたWebページの送信要求に対してレスポンスの悪化を防止することができる。

【0023】

また、本発明は、請求項8に記載されるように、前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するより前又は受信した際に、前記第一の記憶手段が格納する前記書式情報ファイルのうち、使用頻度が高いものから順に所定の数の前記書式情報ファイルを解凍することを特徴とする。

【0024】

このような画像処理装置では、使用頻度が高いものから順に所定の数の書式情報ファイルを解凍しておくため、解凍先の記憶領域の容量を節約することができると共に、要求される頻度の高いWebページの送信要求に対してレスポンスの悪化を防止することができる。

【0025】

また、本発明は、請求項9に記載されるように、前記Webページ生成手段が前記Webページを生成した後、前記Webページの生成に使用された前記書式情報データは、前記第二の記憶手段から削除されることを特徴とする。

【0026】

このような画像処理装置では、Webページの生成に使用した書式情報データは、解凍先の記憶領域から削除されるため、解凍先の記憶領域の容量を節約することができる。

【0027】

また、本発明は、請求項10に記載されるように、前記Webページ生成手段は、要求された前記Webページの生成に使用される前記書式情報データが既に前記第二の記憶手段に格納されているかを判断する解凍済み判断手段を有し、前記Webページ生成手段は、前記解凍済み判断手段の判断結果に基づいて前記書式情報データを使用して前記Webページを生成することを特徴とする。

【0028】

このような画像処理装置では、要求されたWebページの生成に必要な書式情報ファイルが既に解凍されている場合は、解凍された書式情報データを使用してWebページを生成するため、端末に対するレスポンスの悪化を防止することができる。

【0029】

また、本発明は、請求項11に記載されるように、前記Webページ生成手段は、前記書式情報データの数が所定の値を超えた場合は、いずれかの前記書式情報データを前記第二の記憶手段から削除することを特徴とする。

【0030】

このような画像処理装置では、書式情報データの数が所定の値を超えた場合は、いずれかの書式情報データを解凍先の記憶領域から削除するため、解凍先の記憶領域の容量を節約することができる。

【0031】

また、本発明は、請求項12に記載されるように、前記Webページ生成手段は、使用されていない期間が最も長い前記書式情報データを削除することを特徴とする。

【0032】

このような画像処理装置では、使用されていない期間が最も長い書式情報データを削除するため、最近解凍した書式情報データを利用したWebページの要求に対するレスポンスの悪化を防止することができる。

【0033】

また、本発明は、請求項13に記載されるように、前記Webページ生成手段は、解凍

された時期が最も早い前記書式情報データを削除することを特徴とする。

【0034】

このような画像処理装置では、解凍された時期が最も早い書式情報データを削除するため、最近解凍した書式情報データを利用したWebページの要求に対するレスポンスの悪化を防止することができる。

【0035】

更に、上記課題を解決するための手段として、本発明は、上記画像処理装置によって処理されるWebページ生成方法及び上記画像処理装置によって構成される画像処理システムとすることもできる。

【発明の効果】

【0036】

本発明によれば、Webページを生成するための書式情報を圧縮して記憶手段に格納しておくため、書式情報を使用したWebページの生成を少ない記憶容量で行うことができる。

【発明を実施するための最良の形態】

【0037】

以下、図面に基づいて本発明の実施の形態を説明する。図1は、本発明の実施の形態における画像処理システムの構成例を示す図である。図1の画像処理システム1は、画像処理装置10、Webクライアント100a、Webクライアント100b、及びWebクライアント100c（以下、「Webクライアント100」で総称する。）等がLAN（Local Area Network）、イントラネット、又はインターネット等のネットワーク300を介して接続されている。

【0038】

画像処理装置10は、Webクライアント100からの要求に応じて文書データの印刷や、ファックス送信などの処理を実行する。また、画像処理装置10は、Webクライアント100からのHTTPリクエストに応じて、当該HTTPリクエストにおいて要求された情報を表示するWebページをWebクライアント100に対して送信する。例えば、Webクライアント100のユーザがWebブラウザから画像処理装置10のステータス情報を要求した場合は、画像処理装置10は、ステータス情報を格納したHTML（HyperText Markup Language）データをWebクライアント100に送信する。

【0039】

Webクライアント100は、画像処理システム1のユーザが直接利用するPC（Personal Computer）、PDA（Personal Digital(Data) Assistants）、又は携帯電等の端末であり、画像処理装置10から送信されるWebページを表示するWebブラウザを備えている。

【0040】

図2は、本発明の実施の形態における画像処理装置のハードウェア構成例を示す図である。図2の画像処理装置10は、コントローラ60、オペレーションパネル80、FCU81、エンジン部82を有する。

【0041】

コントローラ60は、CPU61、フラッシュメモリ62、ノースブリッジ（NB）63、サウスブリッジ（SB）64、ASIC66、ローカルメモリ67、ネットワークインタフェースカード（NIC）69、USBデバイス70、IEEE1394デバイス71、及びセントロニクス72等を有する。

【0042】

オペレーションパネル80は、コントローラ60のASIC66に接続されている。また、FCU81およびエンジン部82は、コントローラ60のASIC66にPCIバス83で接続されている。

【0043】

コントローラ60は、ASIC66にローカルメモリ67等が接続されると共に、CP

U61とASIC66とがCPUチップセットのNB63を介して接続されている。コントローラ60は、NB63を介してCPU61とASIC66とを接続することにより、CPU61のインタフェースが公開されていない場合に対応する。なお、ASIC66とNB63とはAGP (Accelerated Graphics Port) 65を介して接続されている。

【0044】

このように、画像処理装置10において実行されるアプリケーションを形成する一つ以上のプロセスを実行制御するため、ASIC66とNB63とを低速のPCIBバスでなく、AGP65を介して接続することで、パフォーマンスの低下を防いでいる。

【0045】

CPU61は、画像処理装置10の全体制御を行うと共に画像処理装置10に実装されている各種アプリケーションを起動して実行させる。

【0046】

NB63は、CPU61、フラッシュメモリ62、SB64、ASIC66、NIC69、USBデバイス70、IEEE1394デバイス71およびセントロニクス72を接続するためのブリッジである。

【0047】

SB64、NIC69、USBデバイス70、IEEE1394デバイス71およびセントロニクス72は、PCIBバス73を介してNB63に接続されている。なお、SB64は、PCIBバス73とROMや周辺デバイス等とを接続するためのブリッジである。

【0048】

フラッシュメモリ62は、画像処理装置10において実行される各種アプリケーションや、当該アプリケーションが用いる各種ファイル等が保存されているメモリである。ローカルメモリ67は、アプリケーションの実行時に当該アプリケーションや当該アプリケーションが用いる各種ファイル等がロードされるワーク用のメモリである。

【0049】

ASIC66は、画像処理用のハードウェア要素を有する画像処理用途向けのICである。

【0050】

NIC69は、画像処理装置10をネットワークに接続するインタフェース機器である。USBデバイス70、IEEE1394デバイス71およびセントロニクス72は、夫々の規格に準じたインタフェースである。

【0051】

また、オペレーションパネル80は、オペレータからの入力操作を受け付けると共に、オペレータに向けた表示を行う操作部である。なお、FCU81はバックアップ用のメモリを有している。FCU81が有するメモリは、例えば画像処理装置10の電源がOFFのときに受信したファクシミリデータを一時的に格納するために利用される。

【0052】

図3は、本発明の実施の形態における画像処理装置の機能構成例を示す図である。図3において、画像処理装置10は、スキャナ1231、プリンタ1232及びFAX1233等の個別の画像処理を行うためのハードウェア資源と、エンジン制御ボード1201と、エンジンI/F1202と、OS1203と、API (Application Program Interface) 1204と、スキャナアプリ1211、プリンタアプリ1212、ファックスアプリ1213及びWebアプリ1214等の各種アプリケーションとを有している。

【0053】

エンジン制御ボード1201は、スキャナ1231、プリンタ1232、FAX1233等の画像処理装置10の個別機能を制御する。エンジンI/F1202は、エンジン制御ボード1201に対するインタフェースを、OS1203やAPI1204に提供する。

【0054】

OS1203は、UNIX (登録商標) などのオペレーティング・システムであり、ス

キャナアプリ 1211、プリンタアプリ 1212 等、アプリケーションとして実装されている各種ソフトウェアをプロセスとして並列実行する。

【0055】

API 1204 は、スキャナアプリ 1211、プリンタアプリ 1212 等のアプリケーションへのインタフェースを提供する。

【0056】

スキャナアプリ 1211 は、スキャナ用アプリケーションであり、プリンタアプリ 1212 は、ページ記述言語 (PDL)、PCL 及びポストスクリプト (PS) を有するプリンタ用アプリケーションである。ファックスアプリ 1213 は、ファクシミリ用アプリケーションである。また、Web アプリ 1214 は、画像処理装置 10 にネットワークを介して接続される Web クライアント 100 に対して画像処理装置 10 の機能を提供するためのアプリケーションである。

【0057】

このように、OS 1203 及び API 1204 上に各種アプリケーションを搭載するよう構成しているので、各画像処理の処理部分を共有することが可能となる。従って、画像処理装置 10 全体の生産性を高めるとともに、各アプリケーションの開発効率を高めることができる。

【0058】

次に本発明の第一の実施の形態について説明する。図 4 は、本発明の第一の実施の形態における Web サーバとしての画像処理装置の機能構成例を示す図である。図 4 において画像処理装置 10 は、ROM 11、RAM 12、初期化プログラム 13、解凍プログラム 14、Web サーバプログラム 15、Web アプリケーション 16、及び XSLT プロセッサ 17 等から構成される。

【0059】

ROM 11 は、圧縮された XSL (eXtensible Stylesheet Language) ファイル等を格納するメモリであり、RAM 12 は、解凍プログラム 14 により解凍された XSL データ等を格納するメモリである。ROM 11 は、図 2 におけるフラッシュメモリ 62 に相当し、RAM 12 は、ローカルメモリ 67 に相当する。

【0060】

初期化プログラム 13 は、画像処理装置 10 が起動する際に呼び出され、解凍プログラム 14 を呼び出す。

【0061】

解凍プログラム 14 は、ROM 11 に格納されている圧縮された XSL ファイル (以下、「圧縮 XSL ファイル」という。) を XSL データとして RAM 12 上に解凍する。なお、解凍プログラム 14 は、必ずしも画像処理装置 10 の起動と同時に初期化プログラムから呼び出される必要はなく、Web クライアント 100 から Web サーバプログラム 15 に最初の Web ページの送信要求 (HTTP (HyperText Transfer Protocol) リクエスト) があるまでに呼び出されればよい。従って、例えば後述する Web サーバプログラム 15 がその起動時に解凍プログラム 14 を呼び出してもよい。

【0062】

Web サーバプログラム 15 は、一般に HTTPd (HTTP daemon) と呼ばれるプログラムに相当するものであり、画像処理装置 10 を Web サーバとして機能させるためのものである。Web サーバプログラム 15 は、Web クライアント 100 からの HTTP リクエストに応じて、Web アプリケーション 16 を起動する。また Web サーバプログラム 15 は、Web アプリケーション 16 が生成した HTML (HyperText Markup Language) データを要求元の Web クライアント 100 に送信する。なお、Web サーバプログラム 15 は、図 3 において、API 1204 の一部の機能として画像処理装置 10 に実装されている。

【0063】

Webアプリケーション16は、Webクライアント100に対し、Webページを通して種々のサービスを提供するプログラムである。例えばWebアプリケーション16は、画像処理装置10が管理している文書情報の提供や印刷処理等のサービスを提供する。Webアプリケーション16は、Webサーバプログラム15から呼び出され、サービスの処理結果をXML (eXtensible Markup Language) データとして生成し、更にXSLTプロセッサ17によりXMLデータをHTMLデータ (Webページ) に変換し、当該HTMLデータをWebサーバプログラム15に対して出力する。なお、Webアプリケーション16は、図3において、Webアプリ1214に相当するものである。

【0064】

XSLTプロセッサ17は、Webアプリケーション16から呼び出され、その際に指定されるXMLデータを、RAM上12に格納されているXSLデータを利用して、HTMLデータに変換する。

【0065】

以下、図4の画像処理装置10の処理手順について説明する。図5は、第一の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

【0066】

ステップS10において、画像処理装置10は、起動されると初期化プログラム13を呼び出す。

【0067】

ステップS10に続いてステップS11に進み、初期化プログラム13は解凍プログラム14を呼び出す。ステップS11に続いてステップS12に進み、解凍プログラム14は、解凍すべき全ての圧縮XSLファイル名を確認する。圧縮XSLファイルのファイル名は、例えば所定のファイルに全ての圧縮XSLファイルのファイル名の一覧を管理しておけばよい。その場合、解凍プログラム14はそのファイル (以下「圧縮XSL一覧ファイル」という。) を参照することで全ての圧縮XSLファイルのファイル名を確認することができる。

【0068】

ステップS12に続いてステップS13に進み、解凍プログラム14は、解凍対象とする圧縮XSLファイルのうちの一つをROM11から読み込み、更に解凍後のXSLデータを格納するための領域をRAM12に確保する (S14)。

【0069】

ステップS14に続いてステップS15に進み、解凍プログラム14は圧縮XSLファイルを解凍し、XSLデータとしてRAM12に格納する。ステップS15に続いてステップS16に進み、解凍プログラム14は、解凍対象とする圧縮XSLファイルの全てを解凍したかどうかを判断する。まだ解凍すべき圧縮XSLファイルがROM11にある場合はステップS13に進み、解凍プログラム14は解凍処理を継続する。既に解凍対象とする圧縮XSLファイルの全てを解凍した場合は、解凍処理は終了する。

【0070】

この後、画像処理装置100がWebクライアント100からのHTTPリクエストを受信すると、XSLTプロセッサ17は、RAM12に格納されたXSLデータを利用して、Webアプリケーション16が生成したXMLデータの変換処理を実行する。

【0071】

上述したように第一の実施の形態における画像処理装置10によれば、ROM11には圧縮されたXSLファイルを格納するため、ROM11の容量を節約することができる。また、Webクライアント100からの最初のHTTPリクエストが送信される前に圧縮XSLファイルを解凍しておくため、Webクライアント100に対するレスポンスの悪化を防止することができる。

【0072】

なお、RAM12の容量が少ない等の制約がある場合は、圧縮XSL一覧ファイルに全

ての圧縮XSLファイルのファイル名を登録しておくのではなく、所定のものに限定して登録しておけばよい。この場合、画像処理装置10がWebクライアント100からのHTTPリクエストを受信した時点で、要求されるWebページに対応したXSLデータがRAM12内に存在しない状況が発生するが、その場合は要求されるWebページに対する圧縮XSLファイルをその場で解凍すればよい。要求されたWebページに対応する圧縮XSLファイルの解凍については第三の実施の形態において説明する。また、要求されたWebページに対応する圧縮XSLファイルが既に解凍済みかどうかの判断の方法については第四の実施の形態において説明する。

【0073】

なお、使用頻度の高いものから上位幾つか、又は使用された時期が近いものから上位幾つか等を解凍しておけば、要求されるWebページに対応したXSLデータがRAM12内に存在しない状況の発生の頻度を低く抑えることができる。使用頻度については、圧縮XSL一覧ファイルに各圧縮XSLファイルの使用数のカウンタ（以下「使用数カウンタ」という。）を設け、XSLTプロセッサ17がXSLデータを使用するたびに、そのXSLデータの使用数カウンタをインクリメントすればよい。また、使用された時期については、同じく圧縮XSL一覧ファイルに各圧縮XSLファイルの使用時期を記録する項目を設け、XSLTプロセッサ17がXSLデータを使用するたびに、その使用時期を更新すればよい。解凍プログラム14は、使用数カウンタ又は記録された使用時期を参照することで、使用頻度の高い圧縮XSLファイル、又は使用された時期が近い圧縮XSLファイルを特定することができる。

【0074】

次に本発明の第二の実施の形態について説明する。図6は、本発明の第二の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。第二の実施の形態における画像処理装置20は、第一の実施の形態における画像処理装置10とほぼ同様の構成である。即ち、ROM21はROM11に、RAM22はRAM12に、解凍プログラム24は解凍プログラム14に、Webサーバプログラム25はWebサーバプログラム15に、Webアプリケーション26はWebアプリケーション16に、XSLTプロセッサ27はXSLTプロセッサ17にそれぞれ対応している。但し、画像処理装置20は、初期化プログラム11に相当するものは有しておらず、Webサーバプログラム25が解凍プログラム24の呼び出しを行う。

【0075】

以下、図6の画像処理装置20の処理手順について説明する。図7は、第二の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

【0076】

ステップS20において、画像処理装置20のWebサーバプログラム25は、Webクライアント100からHTTPリクエストを受信すると、解凍済みフラグが1であるかどうかを確認することにより、全ての圧縮XSLファイルが解凍済みであるかどうかを判断する（S21）。

【0077】

解凍済みフラグが1の場合、即ち既に全ての圧縮XSLファイルを解凍済みの場合はステップS28に進む。解凍済みフラグが1でない場合、即ち圧縮XSLファイルの解凍処理を実行していない場合は、ステップS22に進む。

【0078】

ステップS22において、Webサーバプログラム25は解凍プログラム24を呼び出す。ステップS23からステップS27において、解凍プログラム24は第一の実施の形態におけるステップS12からステップS16における手順と同様の手順によってROM21内の全ての圧縮XSLファイルを解凍し、RAM22に格納する。全ての圧縮XSLファイル21の解凍が完了するとステップS28に進む。ステップS28においてWebサーバプログラム25は、解凍済みフラグの値を1にし、ステップS29に進む。

【0079】

ステップS29において、Webサーバプログラム25がWebクライアント100からのHTTPリクエストに対応したWebアプリケーション26を呼び出すと、Webアプリケーション26は所定の処理を実行し、その処理結果としてXMLデータを生成する。ステップS29に続いてステップS30に進み、Webアプリケーション26は、XSLTプロセッサ27を呼び出し、処理結果としてのXMLデータをHTMLデータのWebページに変換し、当該HTMLデータをWebサーバプログラム25に対して出力する。ここでXSLTプロセッサ27は、RAM22上に格納されたXSLデータを利用してXMLデータをHTMLデータに変換する。

【0080】

ステップS30に続いてステップS31に進み、Webサーバプログラム25は、HTMLデータとして生成されたWebページをWebクライアント100に送信する。

【0081】

上述したように、第二の実施の形態における画像処理装置20によれば、ROM21には圧縮されたXSLファイルを格納するため、ROM21の容量を節約することができる。また、画像処理装置20は、Webクライアント100からのHTTPリクエストの受信時であって解凍済みフラグが1でない場合にのみ解凍処理を実行する。ここで、解凍済みフラグは一度解凍処理が実行されると1に設定されるものである。即ち、画像処理装置20は、起動後にWebクライアント100からのHTTPリクエストを最初に受信した際に圧縮XSLファイルを解凍する。従って、画像処理装置20は、Webクライアント100からの最初のHTTPリクエストを受信するまでRAM22の容量を節約できると共に、以降のWebクライアント100からのHTTPリクエストに対するレスポンスの悪化を防止することができる。

【0082】

なお、解凍対象を所定のものに限定してもよく、その実現方法については、第一の実施例の場合と同様である。

【0083】

次に本発明の第三の実施の形態について説明する。図8は、本発明の第三の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。第三の実施の形態における画像処理装置30は、第二の実施の形態における画像処理装置20と同様の構成である。即ち、ROM31はROM21に、RAM32はRAM22に、解凍プログラム34は解凍プログラム24に、Webサーバプログラム35はWebサーバプログラム25に、Webアプリケーション36はWebアプリケーション26に、XSLTプロセッサ37はXSLTプロセッサ27にそれぞれ対応している。但し、画像処理装置30において、解凍プログラム34を呼び出すのは、XSLTプロセッサ37である。

【0084】

以下、図8の画像処理装置30の処理手順について説明する。図9は、第三の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

【0085】

ステップS40において、画像処理装置30のWebサーバプログラム35は、Webクライアント100からHTTPリクエストを受信する。ステップS40に続いてステップS41に進み、Webサーバプログラム35がHTTPリクエストに対応するWebアプリケーション36を呼び出すと、Webアプリケーション36は所定の処理を実行し、処理結果としてXMLデータを生成する。

【0086】

ステップS41に続いてステップS42に進み、Webアプリケーション36は、処理結果としてのXMLデータをHTMLデータに変換すべく、XMLデータとそれに対応するXSLファイルとを指定してXSLTプロセッサ37を呼び出す。ここで、使用するXSLファイルについては、Webアプリケーション36のロジックとして組み込んでおい

てもよい（ハードコーディング）が、Webアプリケーション36が図10に示すXSLファイル対応テーブルを参照することにより判断するようにしてもよい。

【0087】

図10は、圧縮XSLファイル対応テーブルの一例を示す図である。XSL対応テーブル361は、Webクライアント100から要求されたURLである「URL」と、それに対して利用する圧縮XSLファイル名である「XSL」とをデータ項目として有している。Webアプリケーション36は、対応テーブル361を参照することにより、例えばtopPage.cgiのURLが指定された場合は、topPage.xml.zを使用すべきことを判断し、そのファイル名を指定してXSLTプロセッサ37を呼び出す。

【0088】

ステップS42に続いてステップS43に進み、XSLTプロセッサ37は、XSLT変換に必要なXSLデータを手入すべく、Webアプリケーション36から指定された圧縮XSLファイルを指定して、解凍プログラム34を呼び出す。

【0089】

ステップS44及びステップS45において、解凍プログラム34は、第二の実施の形態におけるステップS24及びステップS25における手順と同様の手順で、ROM31内にある指定された圧縮XSLファイルを解凍し、RAM32に格納する。

【0090】

ステップS45に続いてステップS46に進み、XSLTプロセッサ37は、RAM32に格納されたXSLデータを使用してXMLデータをHTMLデータに変換し、Webページを生成する。ステップS46に続いてステップS47に進み、XSLTプロセッサ37は、RAM32において解凍されたXSLデータが格納されている領域を解放することにより、XMLデータをRAM32内から削除する。

【0091】

ステップS47に続いてステップS48に進み、Webサーバプログラム35は、Webクライアント100に対し、生成されたWebページを送信する。

【0092】

上述したように、第三の実施の形態における画像処理装置30によれば、ROM31には圧縮されたXSLファイルを格納するため、ROM31の容量を節約することができる。また、画像処理装置30は、Webクライアント100からのHTTPリクエストを受信した際に、HTTPリクエストの応答としてのWebページの生成に必要な圧縮XSLファイルのみをRAM32に解凍し、Webページの生成が完了したらRAM32におけるXSLデータの領域を解放するため、RAM32の領域を節約することができる。

【0093】

次に本発明の第四の実施の形態について説明する。図11は、本発明の第四の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。第四の実施の形態における画像処理装置40は、第三の実施の形態における画像処理装置30と同様の構成である。即ち、ROM41はROM31に、RAM42はRAM32に、解凍プログラム44は解凍プログラム34に、Webサーバプログラム45はWebサーバプログラム35に、Webアプリケーション46はWebアプリケーション36に、XSLTプロセッサ47はXSLTプロセッサ37にそれぞれ対応している。

【0094】

以下、図11の画像処理装置40の処理手順について説明する。図12は、第四の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

【0095】

ステップS50において、画像処理装置40のWebサーバプログラム45がWebクライアント100からHTTPリクエストを受信すると、以降第三の実施の形態におけるステップS41からステップS43の手順と同様の手順をステップS51からS53において実行し、解凍プログラム44を呼び出す。

【0096】

ステップS53に続いてステップS54に進み、解凍プログラム44は、解凍を要求された圧縮XSLファイルをキャッシュ管理テーブルから検索し、解凍対象の圧縮XSLファイルがキャッシュ管理テーブルに登録されているか否かを判断する。キャッシュ管理テーブルの詳細については後述するが、キャッシュ管理テーブルは、既に解凍されRAM42に格納されている圧縮XSLファイルの一覧を管理しているものである。解凍対象の圧縮XSLファイルがキャッシュ管理テーブルに登録されている場合、即ち既にRAM42に解凍されている場合はステップS58に進み、登録されていない場合、即ちまだ解凍されていない場合はステップS55に進む。

【0097】

ステップS55及びステップS56において、解凍プログラム44は、第三の実施の形態におけるステップS44及びステップ45における手順と同様の手順で、指定された圧縮XSLファイルを解凍する。

【0098】

ステップS56に続いてステップS57に進み、解凍プログラム44は、解凍した圧縮XSLファイルをキャッシュ管理テーブルに登録する。

【0099】

ステップS58以降において、画像処理装置40は、第三の実施の形態におけるステップS47及びS48と同様の手順でWebページを生成し、当該WebページをWebクライアント100に送信する。

【0100】

更に、ステップS54及びS59等によるキャッシュ管理テーブルの検索、更新処理の詳細について説明する。なお、キャッシュ管理テーブルは図13に示すようなリスト構造を利用して実装することができる。

【0101】

図13は、リスト構造によるキャッシュ管理テーブルの実装例を示す図である。図13において、リスト50は、キャッシュエントリ501、502及び503を有している。各キャッシュエントリは、圧縮XSLファイルのファイル名である「ファイル名」と、その圧縮XSLファイルが解凍されたRAM42上のアドレスである「RAM上のアドレス」と、「次のエントリへのポインタ」とをデータ項目として有している。

【0102】

以下、キャッシュ管理テーブルを図13のようなリスト構造で実装した場合のキャッシュ管理テーブルの検索、更新処理の詳細について説明する。図14は、キャッシュ管理テーブルの検索及びLRU法による更新処理を説明するためのフローチャート図である。なお、図14のフローチャートは、図12のフローチャートにおけるステップS54からS57までの処理に該当する。

【0103】

ステップS100において、解凍プログラム44は、リスト50から最初のキャッシュエントリを取り出し、キャッシュエントリのファイル名が解凍対象の圧縮XSLファイルのファイル名と同じかどうか判断する（S101）。ファイル名が同じ場合はステップS102に進み、ファイル名が異なる場合はステップS103に進む。

【0104】

ステップS103において、解凍プログラム44は、既に取り出したキャッシュエントリの「次のエントリへのポインタ」を参照することにより、リスト50から次のキャッシュエントリを取り出す。ステップS103に続いてステップS104に進み、解凍プログラム44は、キャッシュエントリがもう存在しないかどうかを判断する。この判断は、既に取り出したキャッシュエントリの「次のエントリポインタ」の値がNULLか否かによって行うことができる。従って、解凍プログラムは、当該値がNULLだったら次のキャッシュエントリは存在しないと判断する。

【0105】

キャッシュエントリが存在する場合は、ステップS101に進み、再び解凍対象のファイル名とキャッシュエントリのファイル名を比較する。ファイル名が同一の場合は、解凍対象の圧縮XSLファイルは既にRAM42に解凍されているため、ステップS102に進み、現在処理対象のキャッシュエントリをリストの先頭に繋ぎかえて処理を終了する。ここで、処理対象のキャッシュエントリをリストの先頭とするのは、次の検索の際に使用されたXSLデータに対するキャッシュエントリを最初の比較対象とするためである。なお、解凍プログラム44は処理を終了する際に、解凍されたXSLデータのRAM42上におけるアドレスをキャッシュエントリから取り出して、呼び出し元のXSLTプロセッサ47に対して通知する。

【0106】

ステップS104において、リスト50内に解凍対象とする圧縮XSLファイルに対応するキャッシュエントリを発見することができなかった場合は、ステップS105に進み、解凍プログラム44は解凍対象の圧縮XSLファイルをROM41から解凍し、RAM42に格納する。この処理は、図12におけるステップS58及びステップS59に対応する。

【0107】

ステップS105に続いてステップS106に進み、解凍プログラム44は、ステップS105においてRAM42に格納されたXSLデータに対応するキャッシュエントリを生成し、リスト50の先頭に追加する。

【0108】

ステップS106に続いてステップS107に進み、解凍プログラム44は、リスト50のキャッシュエントリの数が所定の上限を超えているかどうかを判断する。この上限はRAM42の容量等に応じて適宜定めればよい。上限を超えていない場合はそのまま処理を終了し、超えている場合は、リスト50の末尾のキャッシュエントリを削除し、さらにその直前のエントリの「次のエントリへのポインタ」の値をNULLにして、処理を終了する。ここで、末尾のキャッシュエントリを削除の対象とするのは、最も長い間利用されていないXSLデータをRAM42から削除するためである。なお、解凍プログラム44は処理を終了する際に、新たに解凍したXSLデータのRAM42上におけるアドレスを、呼び出し元のXSLTプロセッサに対して通知する。

【0109】

図14においては、リスト50においてキャッシュエントリの上限を超えた場合は、最も長い間使用されていないキャッシュエントリが削除、即ち最も長い間使用されていないXSLデータをRAM42から削除するようにしているが（LRU（Least Recently Used）法）、リスト50内において最初に生成されたキャッシュエントリを削除、即ち解凍された時期が最も早いXSLデータをRAM42から削除するようにしてもよい（FIFO（First-In First-Out）法）。その場合のフローチャートを図15に示す。

【0110】

図15は、キャッシュ管理テーブルの検索及びFIFO法による更新処理を説明するためのフローチャート図である。図15において、解凍プログラム44は、解凍対象とする圧縮XSLファイルに対応するキャッシュエントリを見つけた場合には（S201でYesの場合）、そのキャッシュエントリをリスト50の先頭のことなく処理を終了する。また、新たに圧縮XSLを解凍し（S204）、それに対するキャッシュエントリを生成した際には、そのキャッシュエントリはリストの末尾に追加する（S205）。更に、キャッシュエントリ数が上限を超えた場合には（S206でYesの場合）、先頭のキャッシュエントリを削除する（S207）。これにより、FIFO法によりリスト50を更新することができる。

【0111】

上述したように、第四の実施の形態における画像処理装置40によれば、ROM41には圧縮されたXSLファイルを格納するため、ROM41の容量を節約することができる。また、画像処理装置40は、Webクライアント100からのHTTPリクエストを受

信した際に、HTTPリクエストの応答としてのWebページの生成に必要な圧縮XMLファイルのみをRAM42に解凍するため、RAM42の領域を節約することができる。更に、一度解凍した圧縮XMLファイルは、XMLデータとしてRAM42に格納されているため、以降にWebクライアント100からの要求に対して同じXMLデータを利用する場合は、レスポンスの悪化を防止することができる。

【0112】

なお、上述したそれぞれの実施の形態においては、Webサーバとして画像処理装置を適用した例について説明したが、一般的な情報処理装置をWebサーバとして適用しても同様の効果を得ることができる。

【0113】

以上、本発明の好ましい実施例について詳述したが、本発明に係る特定の実施形態に限定されるものではなく、特許請求の範囲に記載された本発明の要旨の範囲内において、種々の変形・変更が可能である。

【図面の簡単な説明】

【0114】

【図1】 本発明の実施の形態における画像処理システムの構成例を示す図である。

【図2】 本発明の実施の形態における画像処理装置のハードウェア構成例を示す図である。

【図3】 本発明の実施の形態における画像処理装置の機能構成例を示す図である。

【図4】 本発明の第一の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。

【図5】 第一の実施の形態における画像処理装置のXMLファイルの解凍処理を説明するためのフローチャート図である。

【図6】 本発明の第二の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。

【図7】 第二の実施の形態における画像処理装置のXMLファイルの解凍処理を説明するためのフローチャート図である。

【図8】 本発明の第三の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。

【図9】 第三の実施の形態における画像処理装置のXMLファイルの解凍処理を説明するためのフローチャート図である。

【図10】 圧縮XMLファイル対応テーブルの一例を示す図である。

【図11】 本発明の第四の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。

【図12】 第四の実施の形態における画像処理装置のXMLファイルの解凍処理を説明するためのフローチャート図である。

【図13】 リスト構造によるキャッシュ管理テーブルの実装例を示す図である。

【図14】 キャッシュ管理テーブルの検索及びLRU法による更新処理を説明するためのフローチャート図である。

【図15】 キャッシュ管理テーブルの検索及びFIFO法による更新処理を説明するためのフローチャート図である。

【符号の説明】

【0115】

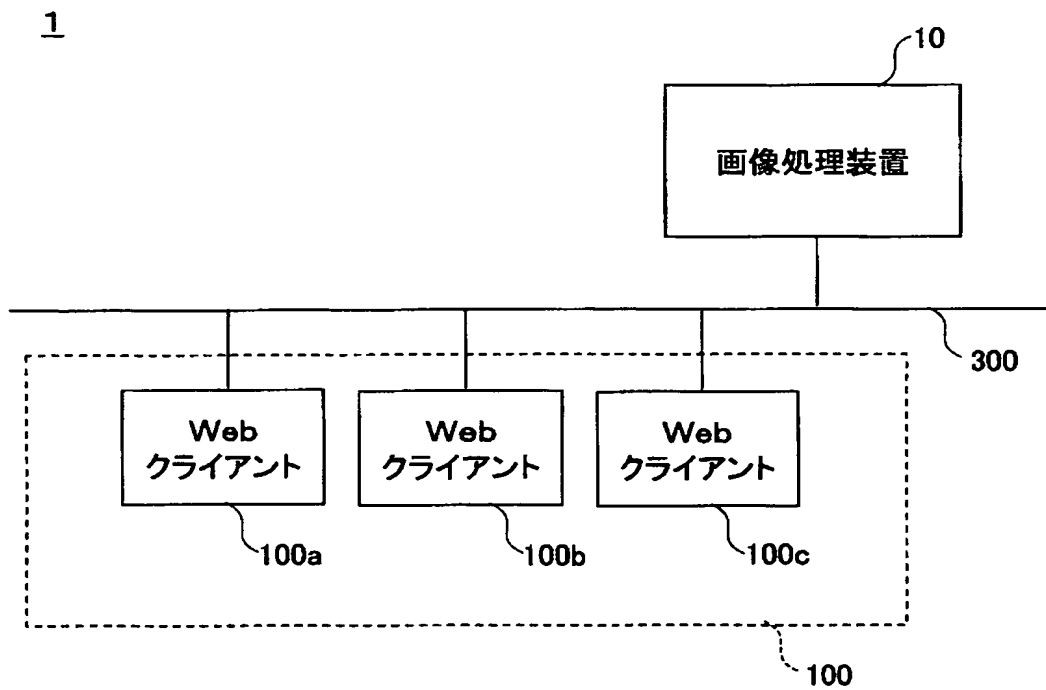
1	画像処理システム
10、20、30、40	画像処理装置
11、21、31、41	ROM
12、22、32、42	RAM
13、23、33、43	初期化プログラム
14、24、34、44	解凍プログラム
15、25、35、45	Webサーバプログラム

16、26、36、46 Webアプリケーション
17、27、37、47 XSLTプロセッサ
50 リスト
60 コントローラ
61 CPU
62 システムメモリ
63 ノースブリッジ (NB)
64 サウスブリッジ (SB)
65 AGP (Accelerated
Graphics Port)
66 ASIC
67 ローカルメモリ
69 NIC
70 USB
71 IEEE1394
72 セントロニクス
73 PCIバス
80 オペレーションパネル
81 ファックスコントロールユニット (FCU)
82 エンジン部
100、100a、100b、100c Webクライアント
300 ネットワーク
361 圧縮XSLファイル対応テーブル
501、502、503 キャッシュエントリ
1201 エンジン制御ボード 1202 エンジンI/F
1203 OS 1204 API
1211 プリンタアプリ 1212 スキャナアプリ
1213 ファックスアプリ 1214 Webアプリ
1231 スキャナ 1232 プリンタ
1233 FAX

【書類名】 図面

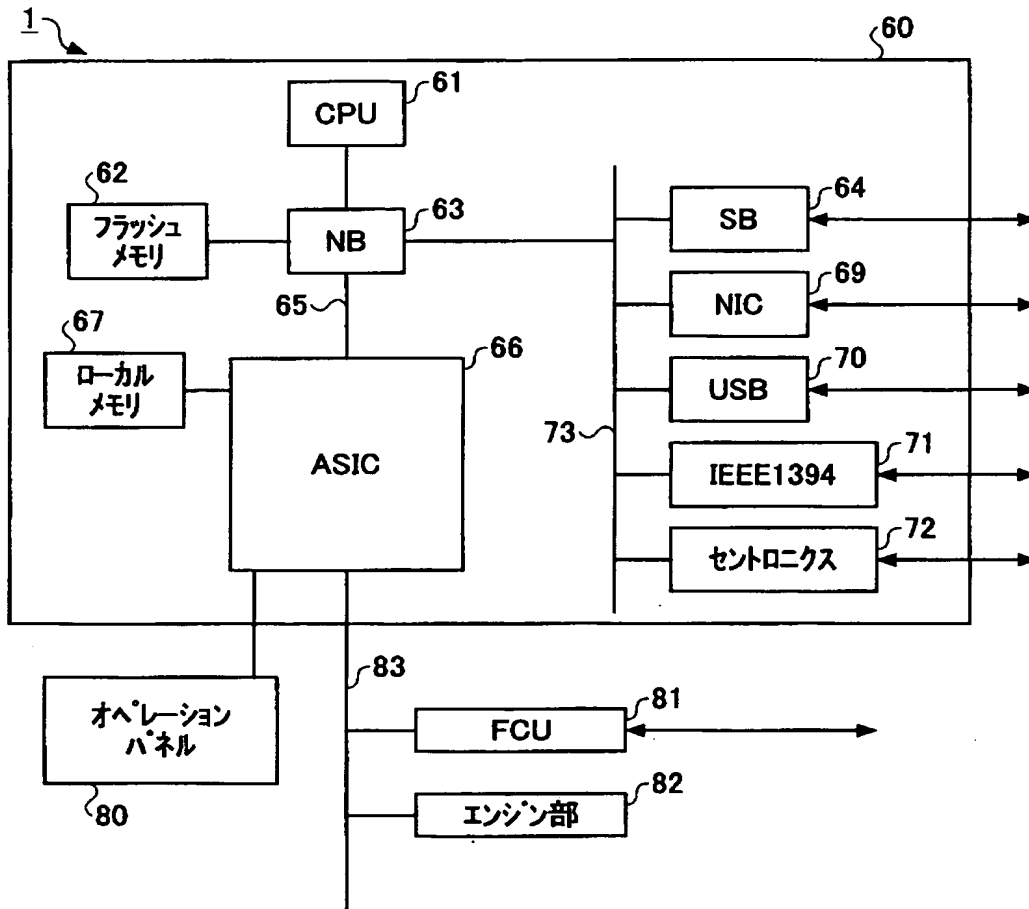
【図 1】

本発明の実施の形態における画像処理システムの
構成例を示す図である



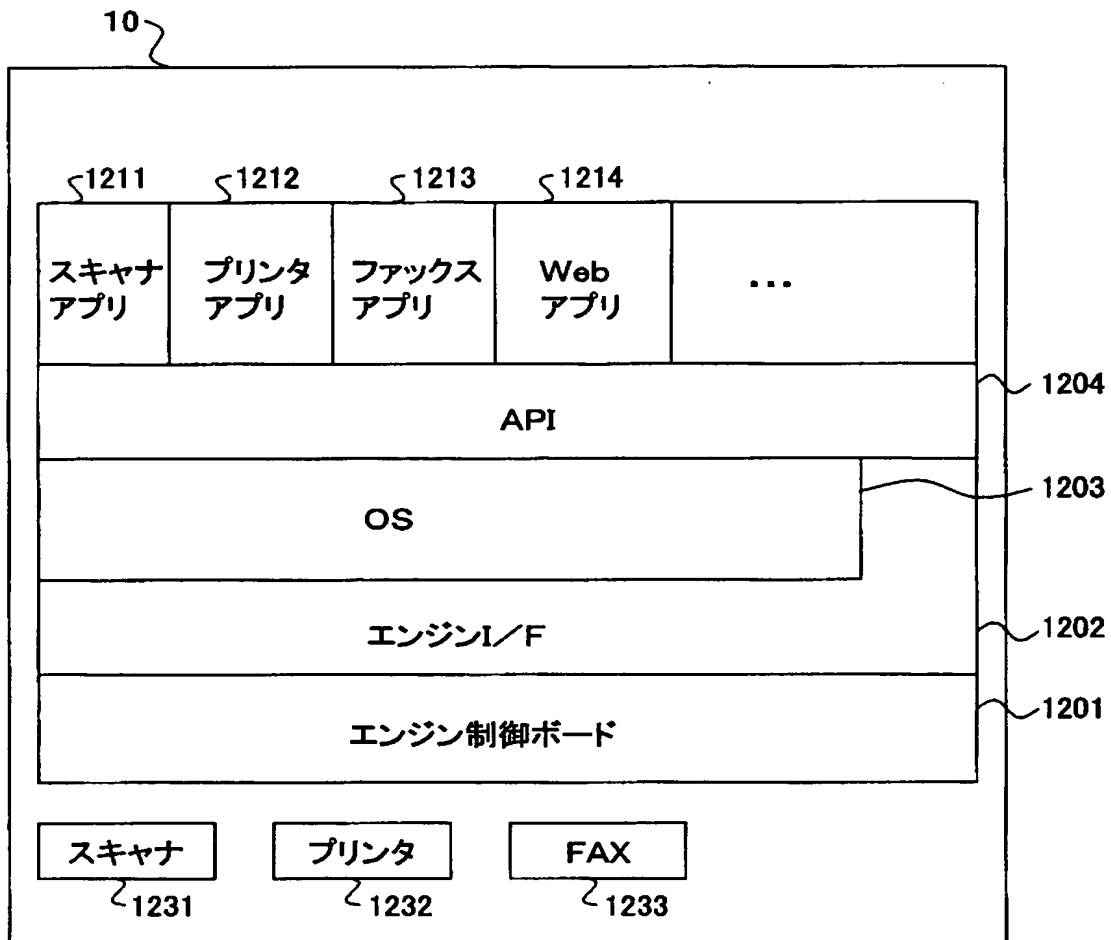
【図 2】

本発明の実施の形態における画像処理装置の
ハードウェア構成例を示す図



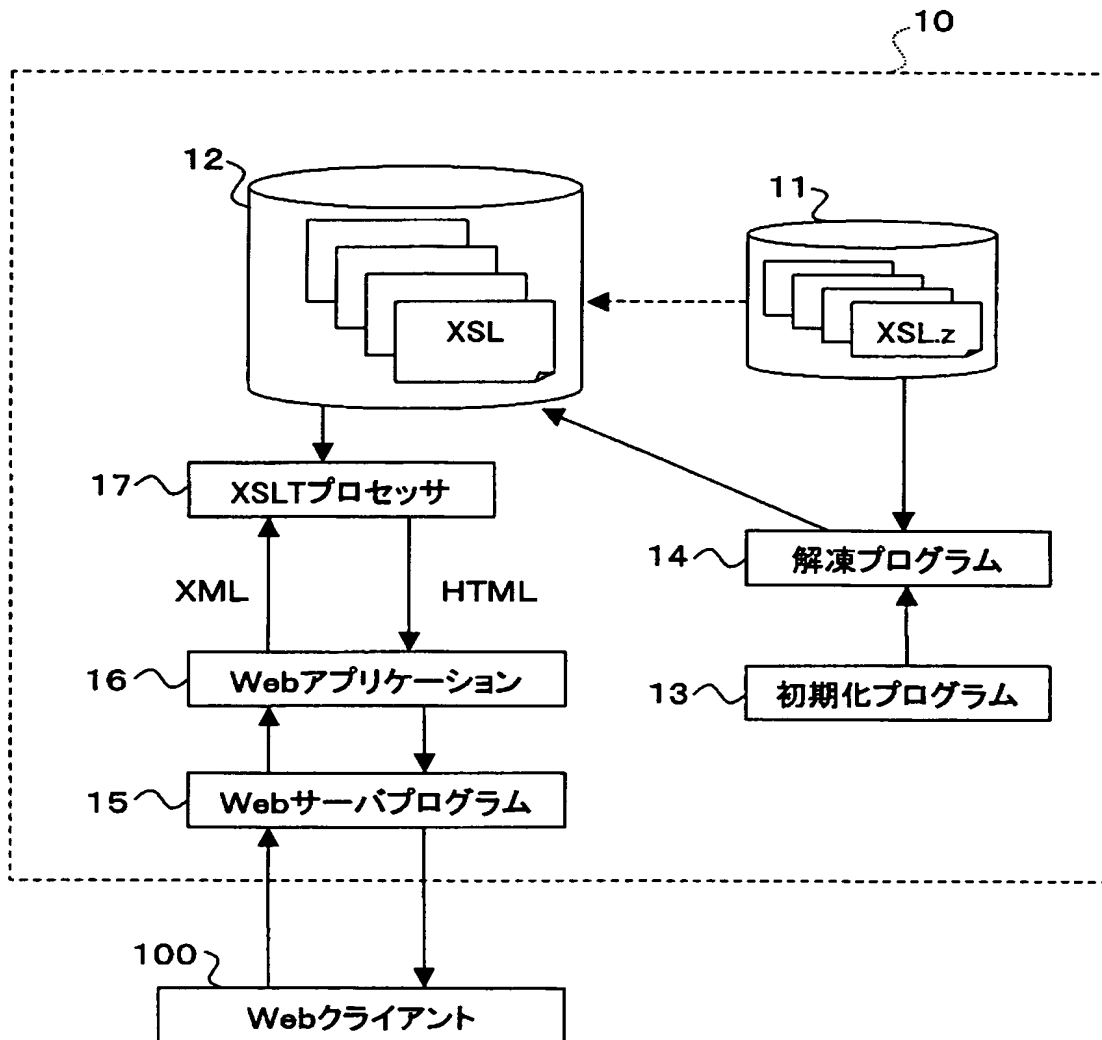
【図 3】

本発明の実施の形態における画像処理装置の機能構成例を示す図



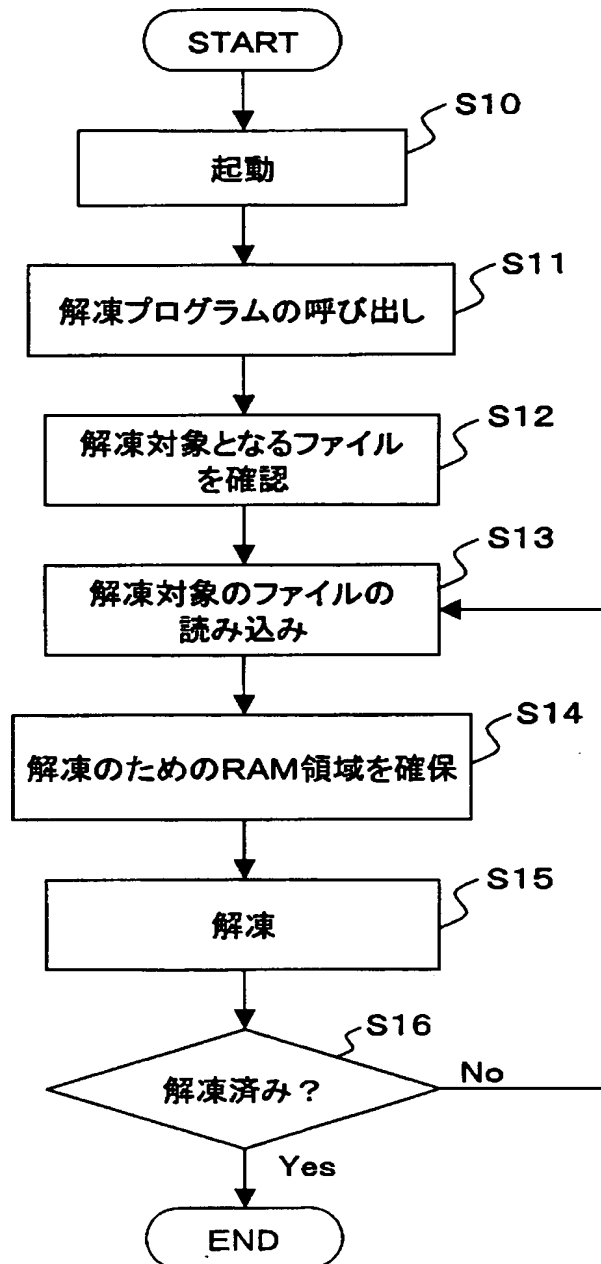
【図 4】

本発明の第一の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図



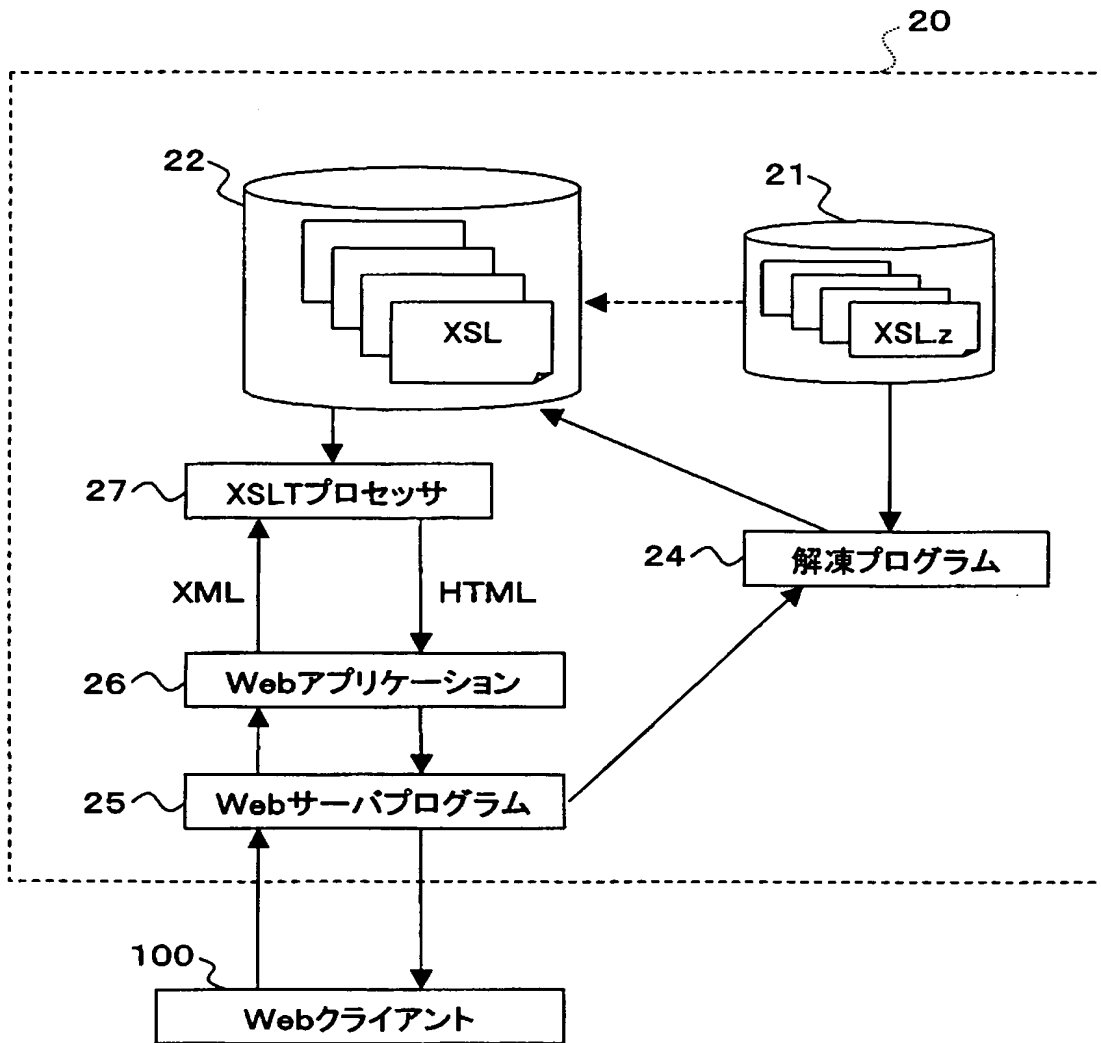
【図 5】

第一の実施の形態における画像処理装置のXSLファイルの
解凍処理を説明するためのフローチャート図



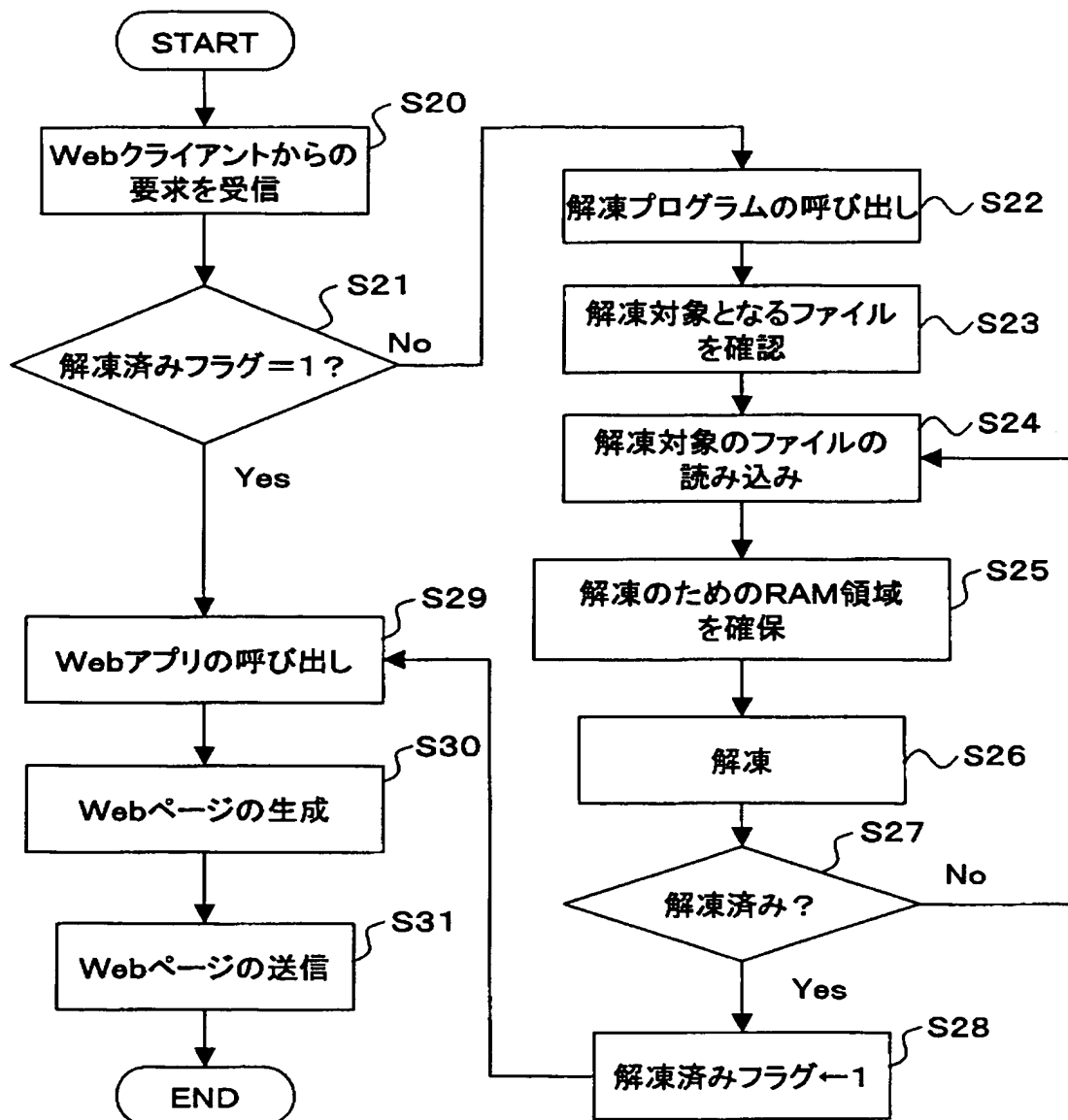
【図 6】

本発明の第二の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図



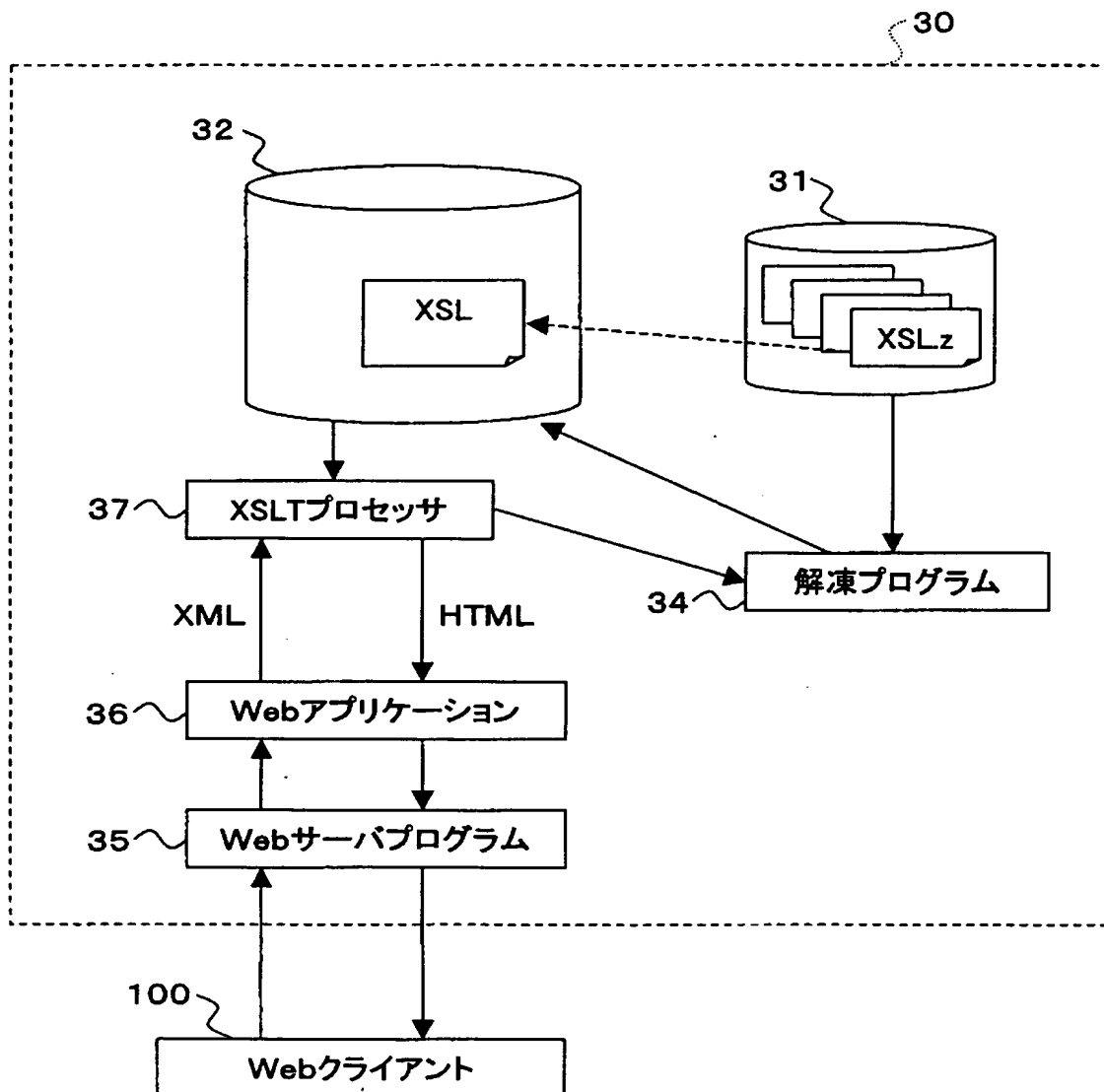
【図 7】

第二の実施の形態における画像処理装置のXSLファイルの
解凍処理を説明するためのフローチャート図



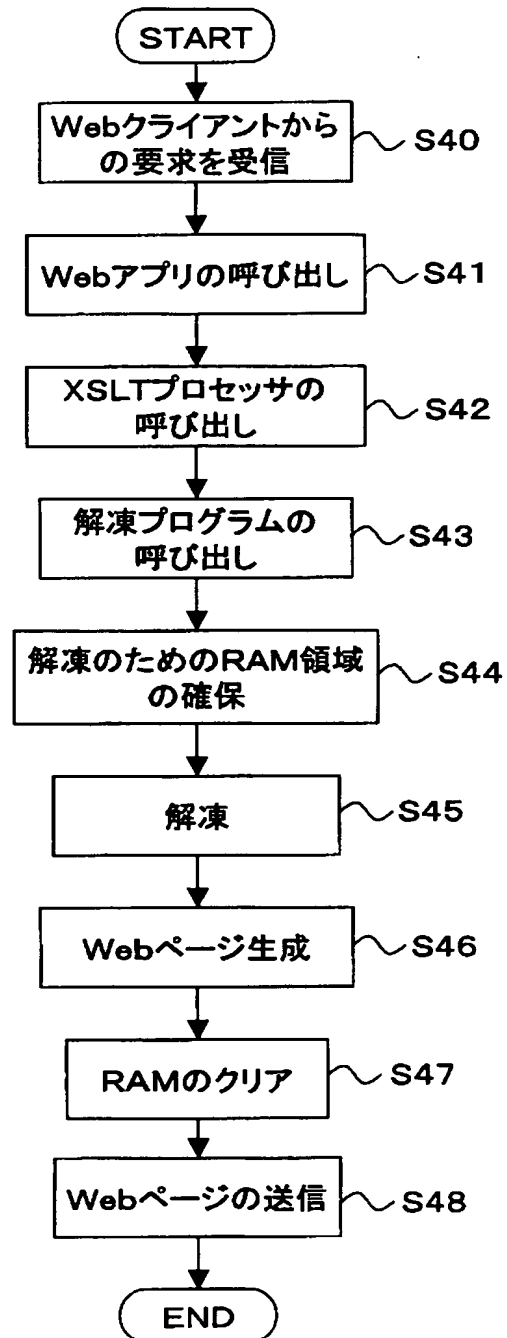
【図 8】

本発明の第三の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図



【図 9】

第三の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図



【図 1 0】

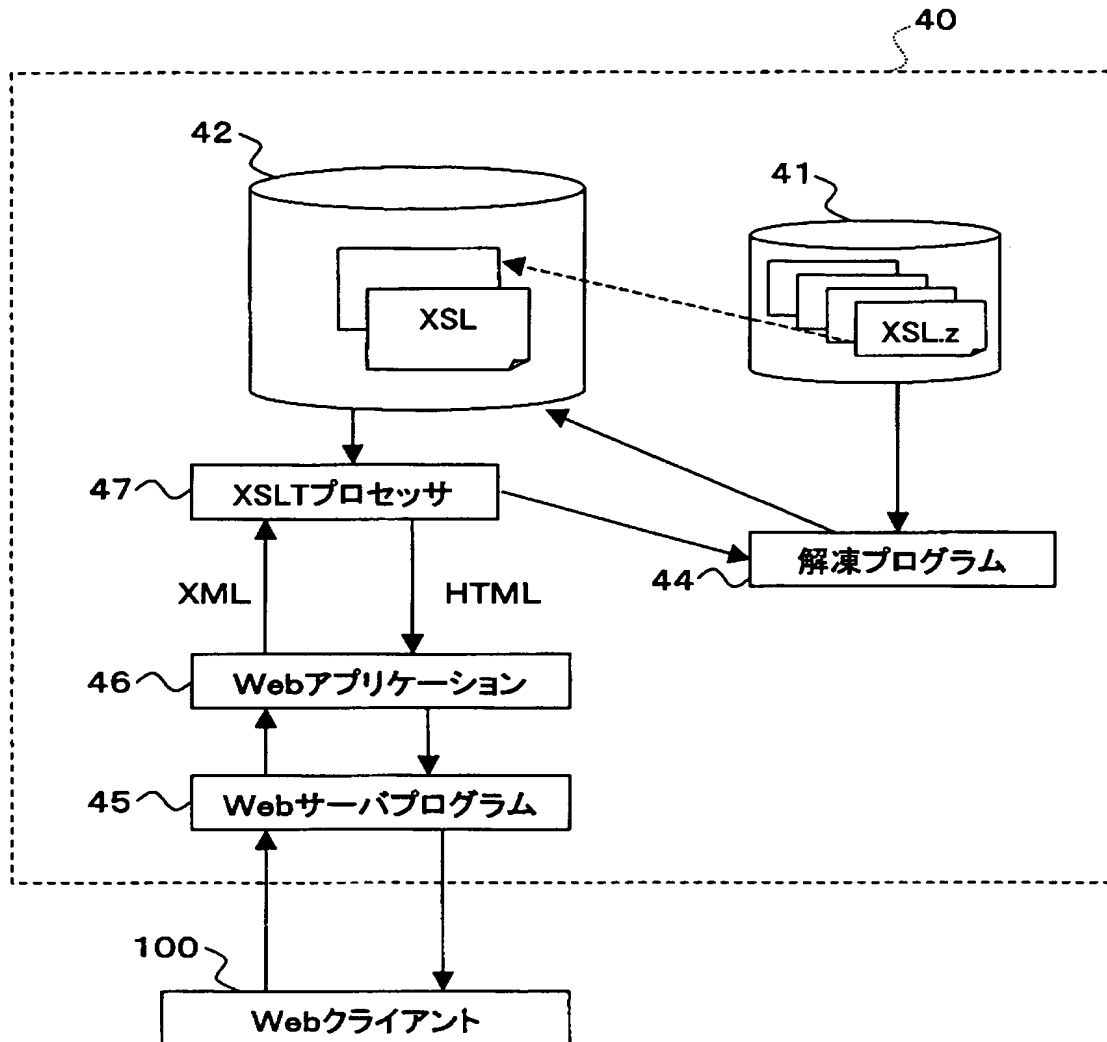
圧縮XSLファイル対応テーブルの一例を示す図

361

URL	XSL
topPage. cgi	topPage. xsl. z
getStatus. cgi	getStatus. xsl. z
inTray. cgi	inTray. xsl. z
outTray. cgi	outTray. xsl. z

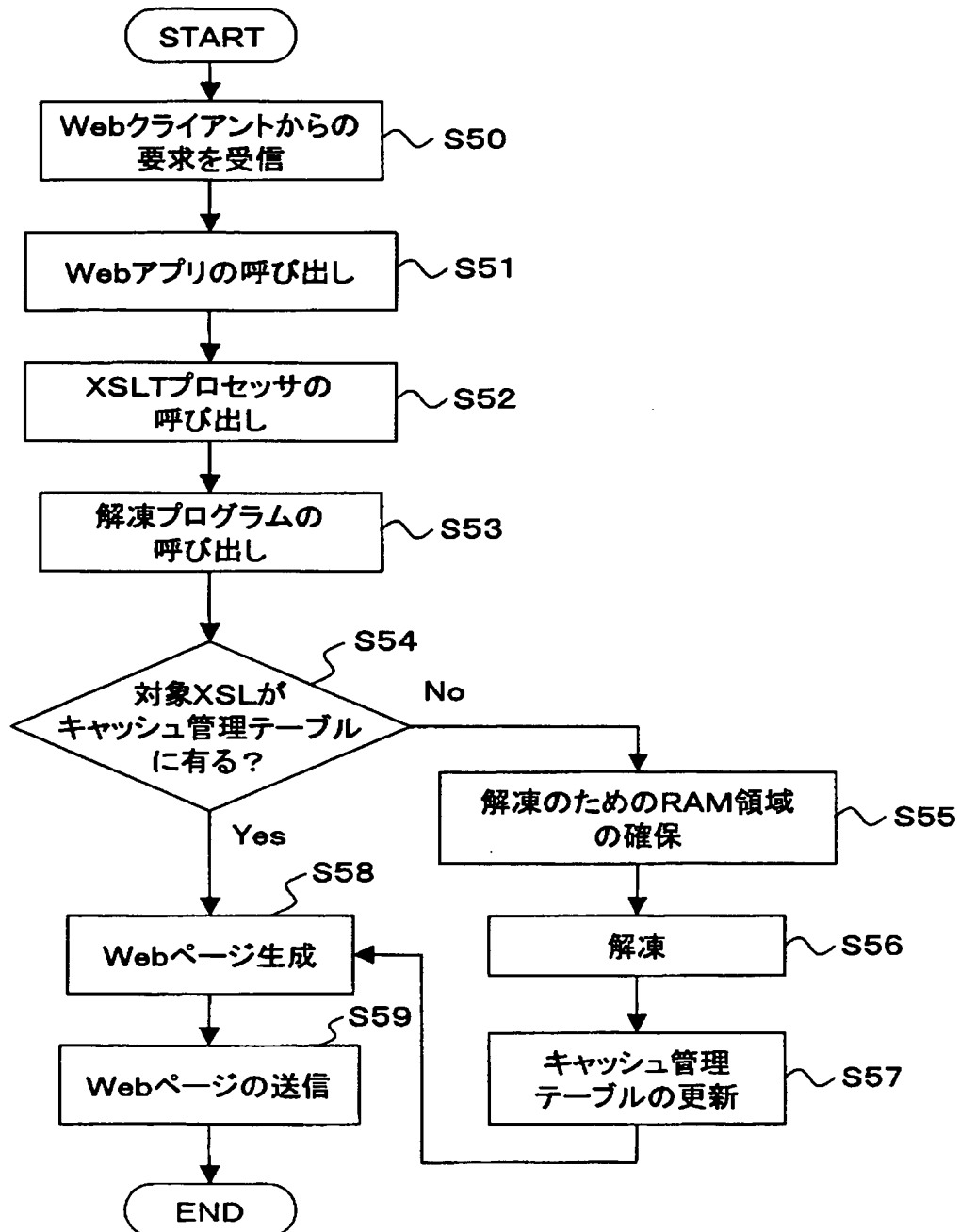
【図 11】

本発明の第四の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図



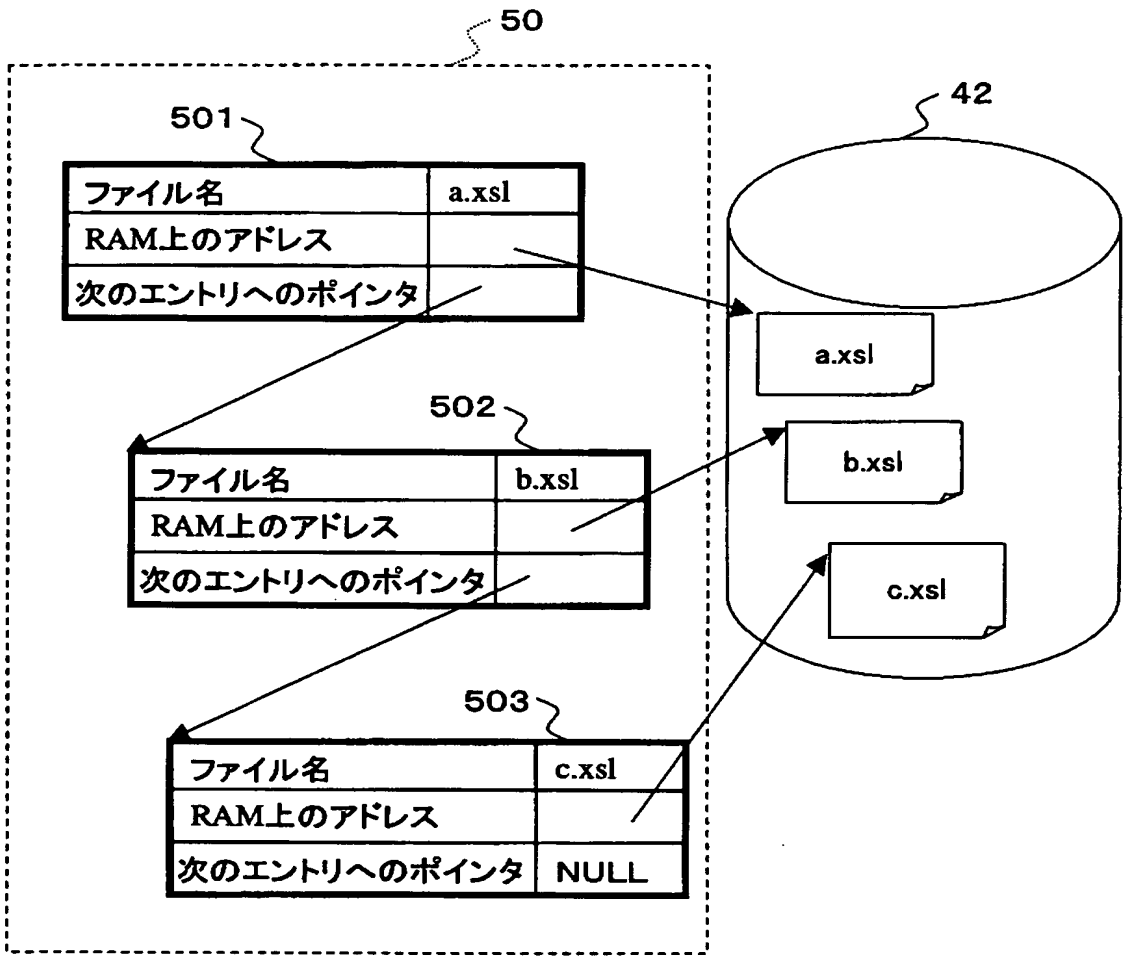
【図 12】

第四の実施の形態における画像処理装置のXSLファイルの
解凍処理を説明するためのフローチャート図



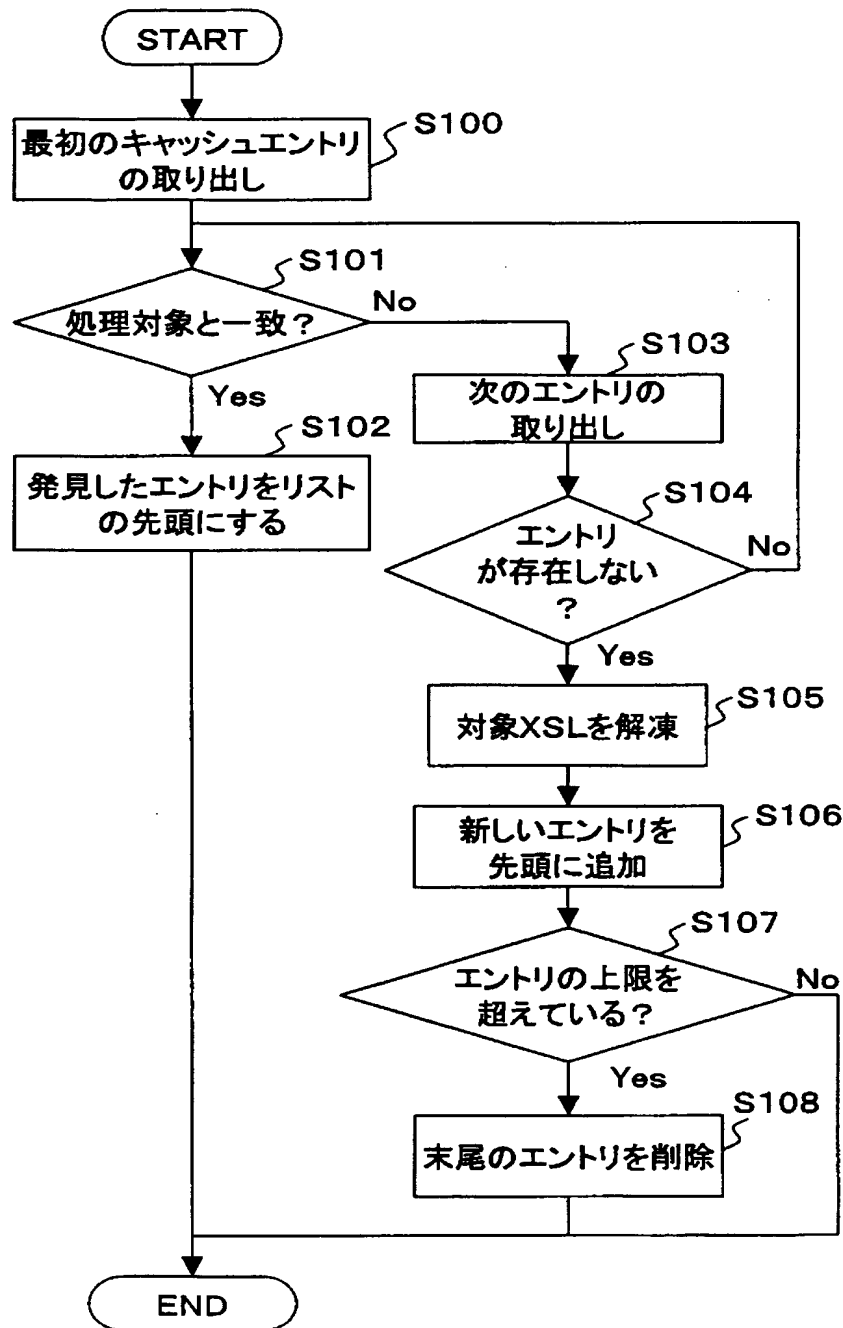
【図 13】

リスト構造によるキャッシュ管理テーブルの実装例を示す図



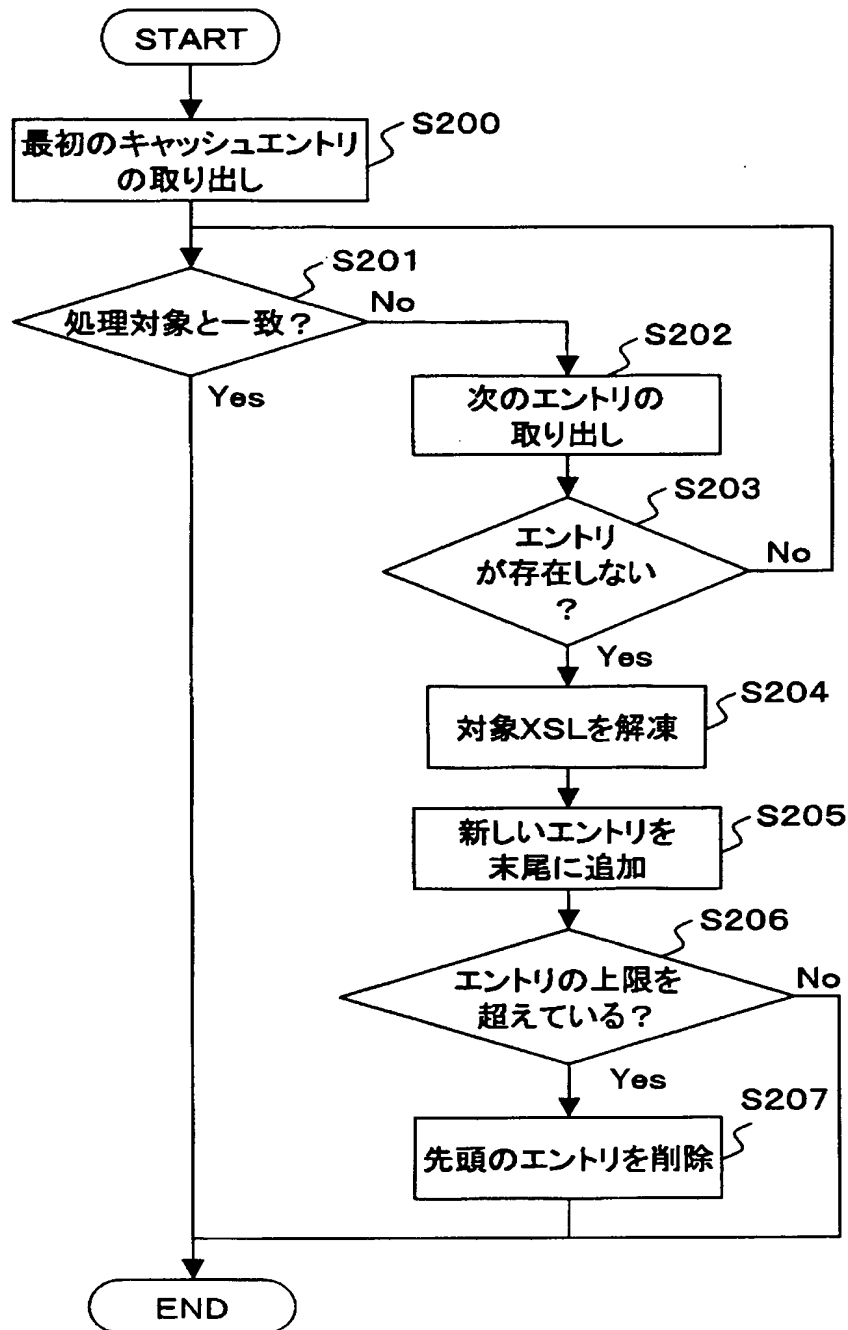
【図 14】

キャッシュ管理テーブルの検索及びLRU法による
更新処理を説明するためのフローチャート図



【図15】

キャッシュ管理テーブルの検索及びFIFO法による
更新処理を説明するためのフローチャート図



【書類名】 要約書

【要約】

【課題】 書式情報を使用したWebページの生成を少ない記憶容量で行うことができる画像処理装置の提供を目的とする。

【解決手段】 ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、前記書式情報データを格納する第二の記憶手段と、前記書式情報データを使用して前記Webページを生成するWebページ生成手段とを有し、該Webページ生成手段により生成された前記Webページを前記送受信制御手段により前記端末に送信することにより上記課題を解決する。

【選択図】 図 4

特願 2 0 0 3 - 2 9 2 4 9 2

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 6 7 4 7]

1. 変更年月日 1 9 9 0 年 8 月 2 4 日
 [変更理由] 新規登録
 住 所 東京都大田区中馬込 1 丁目 3 番 6 号
 氏 名 株式会社リコー

2. 変更年月日 2 0 0 2 年 5 月 1 7 日
 [変更理由] 住所変更
 住 所 東京都大田区中馬込 1 丁目 3 番 6 号
 氏 名 株式会社リコー